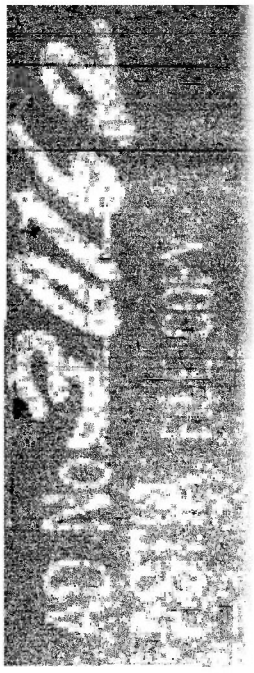
UNCLASSIFIED



MASSACHUSETTS INSTITUTE OF TECHNOLOGY
LINCOLN LABORATORY

CODING FOR CONSTANT-DATA-RATE SYSTEMS
II. MULTIPLE-ERROR-CORRECTING CODES

MARTIN BALSER
RICHARD A. SILVERMAN

23 FEBRUARY 1954

TECHNICAL REPORT NO. 48

UNCLASSIFIED

UNCLASSIFIED

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
LINCOLN LABORATORY

CODING FOR CONSTANT-DATA-RATE SYSTEMS
II. MULTIPLE-ERROR-CORRECTING CODES

Martin Balser
Richard A. Silverman
Group 34

Technical Report No. 48

23 February 1954

ABSTRACT

The study of coding for constant-data-rate systems, begun in Part I, is extended by considering the use of multiple-error-correcting codes. The principle of the Wagner code is used to construct two new multiple-error-correcting codes, the Hamming-Wagner code and the syllabified Wagner code. The performances of these codes, and of the Reed code (a multiple-error-correcting code not of the Wagner type) are compared. Of the three Wagner-type codes, the ordinary Wagner code is best for short words, the Hamming-Wagner for medium length and long words, and the syllabified Wagner for very long words. The Reed three-error-correcting code (as yet only applicable to a few isolated values of word length) outperforms both the Hamming-Wagner and syllabified Wagner codes.

CAMBRIDGE MASSACHUSETTS

UNCLASSIFIED

CODING FOR CONSTANT-DATA-RATE SYSTEMS
II. MULTIPLE-ERROR-CORRECTING CODES

A. INTRODUCTION

In Part I of this paper[1] we introduced the Wagner code, a new means of correcting single errors in sequences of binary digits. (We call such sequences words.) It differs from the Hamming code[2] by being likely rather than certain to correct single errors. However, it uses only one check digit, whereas the Hamming code uses several (the number depending on the number of message digits). In communication systems with a fixed word length (constant-data-rate) the economy of the Wagner code in the use of check digits can offset the disadvantage of not correcting all single errors.* We found that for such systems short Wagner-coded words have much smaller probabilities of error than the corresponding Hamming-coded words.

We now consider the performance of multiple-error-correcting codes in constant-data-rate systems. Again we have codes like Hamming's, which correct errors by using algebraic relations between received message and check digits, and codes like Wagner's, which require stored a posteriori probabilities suitably computed by the receiver, as well as the received word.[1] (Some may object to the word "code" as applied to the Wagner scheme, since information other than the received word is required. However, we feel that the phrase "Wagner code" is justified by its linguistic convenience.) There are also "mixed" codes, which correct some errors by using only algebraic relations between the received digits, and other errors by using the Wagner scheme. The Hamming-Wagner code described in Sec. B is such a "mixed" code. We also examine a syllabified Wagner code, in which each word is split up into separately Wagner-coded subwords, and the class of multiple-error-correcting codes recently developed by I. S. Reed and others.[3] Our conclusions are summarized in Sec. E.

B. THE HAMMING-WAGNER CODE

We consider systems such that each digit (one of two electrical signals, $x_1(t)$ and $x_2(t)$ of duration T and bandwidth W, TW >> 1) is corrupted in the channel by the addition of white Gaussian noise. If y(t) is the received signal, $x_1(t)$ or $x_2(t)$ is chosen as the transmitted signal according as

$$z_1 = \int_0^T x_1(t)y(t)dt \tag{1}$$

or

$$z_2 = \int_0^T x_2(t)y(t)dt \tag{2}$$

is the larger. It was shown in Part I, with which familiarity is assumed, that the correlations z_1 and z_2 are monotone functions of the a posteriori probabilities $p(x_1/y)$ and $p(x_2/y)$.** Moreover, they are more convenient quantities for calculation, since for suitable x_1 and x_2, they are

*The reader is reminded that the shorter a digit, the greater its probability of error.[1]

**The a posteriori probability that if y is received, x was sent, is denoted by p(x/y).

independent Gaussian random variables, with means c_1 and c_2 ($c_1 > c_2$), and standard deviations σ_1 and σ_2.

The Wagner code (analyzed in Part I) operates as follows: the values of z_1 and z_2 corresponding to each digit of a received word are stored in a memory for the duration of the word. The last digit of each transmitted word was chosen to make the sum of all the digits even (parity check). The sum of all the tentatively identified digits will be odd if the received word differs from the corresponding transmitted word in an odd number of digits. However, whenever the sum is odd, the receiver assumes that the error is in only one digit, and alters the digit for which the stored correlations differ by the smallest amount. We found that the probability of error per Wagner-coded word of m message digits is

$$P_W = 1 - q^{m+1}(a) - \Pi_{m+1}(a) \quad , \tag{I-29}^*$$

where

$$a = \frac{c_1 - c_2}{\sqrt{2(\sigma_1^2 + \sigma_2^2)}} \quad , \tag{3}$$

and

$$p(a) = \tfrac{1}{2}(1 - \operatorname{erf} a) \quad , \tag{I-7}$$

$$q(a) = \tfrac{1}{2}(1 + \operatorname{erf} a) \quad . \tag{I-14}$$

The quantity $\Pi_n(a)$ is the multiple integral

$$\Pi_n(a) = \frac{n!}{(\sqrt{\pi})^n} \int_0^\infty \exp[-(x_n - a)^2]dx_n \int_0^{x_n} \exp[-(x_{n-1} - a)^2]dx_{n-1} \cdots$$

$$\int_0^{x_3} \exp[-(x_2 - a)^2]dx_2 \int_0^{x_2} \exp[-(x_1 + a)^2]dx_1 \quad , \tag{I-12}$$

which can be reduced by repeated integration by parts to the recurrence relation

$$\Pi_n(a) = \tfrac{1}{2}\binom{n}{n-1}\Pi_{n-1}(a) - \frac{1}{2^2}\binom{n}{n-2}\Pi_{n-2}(a) \cdots + \frac{(-1)^{n-1}}{2^{n-2}}\binom{n}{2}\Pi_2(a)$$

$$+ \frac{(-1)^n}{2^n}\binom{n}{1}[I_1(a) - I_n(a)] \quad , \tag{I-22}$$

where

$$I_n(a) = \frac{2}{\sqrt{\pi}} \int_0^\infty [\operatorname{erf}(x - a)]^{n-1} \exp[-(x + a)^2]dx \quad . \tag{I-23}$$

---

*Equation numbers preceded by the Roman numeral I refer to correspondingly numbered equations in Part I.[1]

It is shown in Appendix B that Eq. (I-22) can be reduced to the sum

$$\Pi_n(a) = \frac{n}{2^n} \sum_{i=1}^{n} \binom{n-1}{i-1}(-1)^{i+1} I_i(a) \quad . \tag{4}$$

All the terms of Eq. (4) are positive for values of a in the range of interest[1] (a = 1.0 to 3.0, say), so that Eq. (4) is much more suited for numerical work than Eq. (I-22) when n is large.

We now extend the principle of the Wagner code to a double-error-correcting code. The following procedure appears best as a first attempt. Further check digits are added to the Wagner-coded word; these reveal double as well as single errors.* If a double error is detected, we change the <u>two</u> digits of the stored word with the smallest correlator differences. If a single error is detected, we change only the smallest correlator difference.

The success of this scheme requires a system of check digits which indicates both single and double errors, and further allows them to be <u>distinguished</u>. The geometrical model of message space (see Appendix A) is well suited for examining the possibility of setting up such check digits. Referring to Fig. 1, we see that if both single and double errors in possible transmitted points (such as P_1 and P_2) are to be detectable, and if single errors are to be distinguishable from double errors, every such pair of points must be separated by a distance of 4 or more. For then, a single error in P_1 sends it to a neighboring point like S_1, where it can be stated with certainty to have come either from P_1 by a change in one digit, or from some other possible transmitted message by a change in three or more digits. Similarly, a single error in P_2 sends it to a neighbor like S_2. On the other hand, a double error in either P_1 or P_2 may correspond to a received point like D, at a distance of 2 from both. Unless there are at least three points between all pairs of possible transmitted points, a double error in P_1 (say) is indistinguishable from a single error in P_2 (or some other transmitted point), so that we do not know whether to correct one or two digits in the received word.

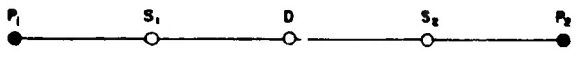


Fig. 1. Configuration of points in message space between two possible transmitted messages P_1 and P_2.

Now in a Hamming single-error-correcting, double-error-detecting code, all transmitted messages are separated by at least a distance of 4 (see Appendix A). This is just the separation required for successful operation of a Wagner code that corrects both single and double errors. Thus the number of check digits needed to correct <u>all</u> single errors before applying the Wagner procedure to double errors is the same as the number required to apply the Wagner procedure to both single and double errors. This suggests a "Hamming-Wagner" code of the "mixed" type mentioned in Sec. A, which is obviously better than the corresponding "Wagner-Wagner" code.

We thus arrive at a code that is like the Hamming single-error-correcting, double-error-detecting code, except that if the extra check digit indicates a double error, we change the <u>two</u> digits with the smallest correlator differences. The analysis of this Hamming-Wagner

*It is inevitable that some higher-order errors will be mistaken for double errors and will remain uncorrected after applying the Wagner procedure. However, this is no worse than leaving them uncorrected in the first place. (See Appendix A.)

code is completely analogous to that of the simple Wagner code.

The probability of error per Hamming-Wagner-coded word is

$$P_{HW} = 1 - q^{m+k+1}(a) - (m + k + 1)\, q^{m+k}(a)\, p(a) - {}^2\Pi_{m+k+1}(a) \quad , \tag{5}$$

where a, p (a), and q (a) have already been defined. The quantity k is the number of check digits required by the Hamming single-error-correcting code.* The quantity ${}^2\Pi_n(a)$ [in analogy to Eq. (I-12)] is the multiple integral

$${}^2\Pi_n(a) = \frac{n!}{(\sqrt{\pi})^n} \int_0^\infty \exp[-(x_n - a)^2]dx_n \int_0^{x_n} \exp[-(x_{n-1} - a)^2]dx_{n-1} \cdots$$

$$\cdots \int_0^{x_4} \exp[-(x_3 - a)^2]\,dx_3 \int_0^{x_3} \exp[-(x_2 + a)^2]dx_2 \int_0^{x_2} \exp[-(x_1 + a)^2]dx_1 \quad . \tag{6}$$

Repeated integration by parts reduces Eq. (6) to the recurrence relation

$${}^2\Pi_n(a) = \tfrac{1}{2}\binom{n}{n-1}\,{}^2\Pi_{n-1}(a) - \frac{1}{2^2}\binom{n}{n-2}\,{}^2\Pi_{n-2}(a) \cdots$$

$$+ \frac{(-1)^{n-2}}{2^{n-3}}\binom{n}{3}\,{}^2\Pi_3(a) + \frac{(-1)^{n-1}}{2^{n-1}}\binom{n}{2}[{}^2I_2(a) - {}^2I_n(a)] \quad , \tag{7}$$

where

$${}^2I_n(a) = \frac{2}{\sqrt{\pi}} \int_0^\infty [\operatorname{erf}(x - a)]^{n-2} \exp[-(x + a)^2][\operatorname{erf}(x + a) - \operatorname{erf} a]dx \quad . \tag{8}$$

Equation (7) can be further reduced to the sum

$${}^2\Pi_n(a) = \frac{n(n-1)}{2^n} \sum_{i=2}^{n} \binom{n-2}{i-2}(-1)^i\, {}^2I_i(a) \tag{9}$$

in complete analogy to Eq. (4). (See Appendix B.)

$P_{HW}(1.35)$ and $P_{HW}(1.80)$ are tabulated in Table I for various values of m, together with the corresponding probabilities of error for uncoded, Hamming-coded, and Wagner-coded words. The values of a used in computing P_U, P_H, and P_W are chosen so that all words (message digits plus check digits) have the same duration, as required in a constant-data-rate system. Thus

$$P_U(a_U) = 1 - q^m(a_U) \quad , \qquad a_U = \sqrt{\frac{m + k + 1}{m}}\, a \tag{10}$$

$$P_H(a_H) = 1 - q^{m+k}(a_H) - (m + k)\, q^{m+k-1}(a_H)\, p(a_H) \quad , \qquad a_H = \sqrt{\frac{m + k + 1}{m + k}}\, a \tag{11}$$

$$P_W(a_W) = 1 - q^{m+1}(a_W) - \Pi_{m+1}(a_W) \quad , \qquad a_W = \sqrt{\frac{m + k + 1}{m + 1}}\, a \tag{12}$$

*For m = 5 to 11, k = 4; for m = 12 to 26, k = 5; etc.[2]

TABLE I

COMPARISON OF HAMMING, WAGNER, AND HAMMING-WAGNER CODES

(a) $\alpha = 1.35$

m	P_U	P_H	P_W	P_{HW}
10	0.093	0.044	0.030	0.037
11	0.111	0.050	0.038	0.043
12	0.110	0.065	0.038	0.057
13	0.128	0.073	0.047	0.064
14	0.146	0.081	0.056	0.071
15	0.165	0.090	0.067	0.079
16	0.183	0.098	0.079	0.087
17	0.202	0.107	0.092	0.096
18	0.220	0.116	0.105	0.104
19	0.239	0.126	0.119	0.113
20	0.257	0.135	0.134	0.122
21	0.275	0.145	0.149	0.132

(b) $\alpha = 1.80$

m	P_U	P_H	P_W	P_{HW}
10	0.0091	0.0016	0.00067	0.00063
11	0.0117	0.0018	0.00095	0.00076
12	0.0109	0.0025	0.00081	0.00107
13	0.0135	0.0029	0.00111	0.00124
14	0.0163	0.0033	0.0015	0.0014
15	0.0193	0.0037	0.0019	0.0016
16	0.0224	0.0042	0.0024	0.0019
17	0.0258	0.0046	0.0030	0.0021
18	0.0292	0.0051	0.0037	0.0024
19	0.0328	0.0057	0.0044	0.0026
20	0.0364	0.0062	0.0052	0.0029
21	0.0401	0.0068	0.0061	0.0032
22	0.0439	0.0074	0.0070	0.0036
23	0.0478	0.0080	0.0080	0.0039
24	0.0518	0.0086	0.0091	0.0043

Values of α are for the Hamming-Wagner code
m = number of m ssage digits

Table I shows that for $\alpha = 1.35$, a very noisy case, the Hamming code becomes better than the Wagner code at m = 21. For $\alpha = 1.80$, which corresponds to much less noise, the Hamming code surpasses the Wagner code at m = 24. Thus it appears that, starting with some value of m between 25 and 30, the Hamming code is better than the Wagner code anywhere in the significant range of α (neither too little nor too much noise[1]). This happens for the reasons given in Part I: – (1) the ratio k/m decreases with increasing m, so that corresponding values of α for the two codes become more nearly alike, dissipating the advantage of the Wagner code's economy in the use of check digits, and (2) the conditional probability that the Wagner code corrects single errors decreases as m increases.

We see from the table that the Hamming-Wagner code is consistently better than the Hamming code; however, the percentage improvement is greater for $\alpha = 1.80$ than for the noisier case $\alpha = 1.35$. For $\alpha = 1.35$, the Hamming-Wagner code is better than the Wagner code for all $m > 17$; for $\alpha = 1.80$, the Hamming-Wagner code is better than the Wagner code for all $m > 13$.* Thus, while the Wagner code is superior to the Hamming code for words of length less than about 20,** the Hamming-Wagner code is superior to either of these codes for words of length greater than about 15.† The Hamming-Wagner code works better in low noise than in high noise, because (1) proportionately fewer multiple errors are of order higher than two, and (2) the conditional probability of correcting double errors is higher. Since this conditional probability decreases as m increases, the Hamming-Wagner code gradually becomes less effective, as shown in the next section.

C. THE SYLLABIFIED WAGNER CODE

Another multiple-error-correcting code based on the principle of the Wagner code is the syllabified Wagner code, constructed by dividing each word into separately Wagner-coded subwords or syllables. Suppose a word with m message digits is divided into j syllables, each containing $n_i = m_i + 1$ digits, where

$$m = \sum_{i=1}^{j} m_i \quad .$$

Since the probability that a syllable (regarded as a Wagner-coded word) is correct is

$$q^{n_i}(\alpha) + \Pi_{n_i}(\alpha) \quad ,$$

the probability of error for a syllabified-Wagner-coded word is

$$P_{SW}(m_1, m_2, \ldots, m_j) = 1 - \prod_{i=1}^{j} [q^{n_i}(\alpha) + \Pi_{n_i}(\alpha)] \quad , \quad \sum_{i=1}^{j} m_i = m \quad . \qquad (13)$$

*The Hamming-Wagner code is also better for m = 10 and 11. This anomaly is due to the change in k from 4 to 5 at m = 12.

**A comparison of the Hamming and Wagner codes in the range m = 4 to 8 is given in Part I.

†The value of m for which one code becomes better than another is somewhat dependent on α. (See Table I.)

It follows at once that for a given number of syllables $P_{SW}(m_1, m_2, \ldots, m_j)$ is smallest when the syllables have equal length (or as nearly equal as possible). For if we write Eq. (13) as

$$P_{SW}(m_1, m_2, \ldots, m_j) = 1 - f\left(m - \sum_{i=1}^{j-1} m_i\right) \prod_{i=1}^{j-1} f(m_i) \quad , \tag{14}$$

where

$$f(m_i) = q^{n_i}(\alpha) + \Pi_{n_i}(\alpha) \quad , \tag{15}$$

we obtain after differentiating Eq. (14) with respect to m_k and equating the result to zero

$$\frac{f'(m_k)}{f(m_k)} = \frac{f'\left(m - \sum_{i=1}^{j-1} m_i\right)}{f\left(m - \sum_{i=1}^{j-1} m_i\right)} \quad , \quad k = 1, 2, \ldots, j-1 \quad . \tag{16}$$

Consequently,

$$\frac{f'(m_1)}{f(m_1)} = \frac{f'(m_2)}{f(m_2)} = \ldots = \frac{f'(m_j)}{f(m_j)} \quad , \tag{17}$$

so that $P_{SW}(m_1, m_2, \ldots, m_j)$ is smallest when all the $f(m_i)$ are equal, i.e., when all the syllables are of equal length (or as nearly equal as possible).

If too few syllables are used, the conditional probability of correction of single errors per syllable is small because the syllables are too long. If too many syllables are used, this conditional probability is small because the large number of check digits leads to a small value of α. (This second effect is partially compensated by increased multiple-error-correction possibilities.) The optimum number of syllables is a compromise between these two effects. This optimum number is not necessarily critical, or for that matter the same for all α. The simple Wagner code (which may be considered a syllabified Wagner code of one syllable) is clearly best for short words. At about m = 14, division into two syllables is better than the simple Wagner code. At m = 30, divisions into three and four syllables are about equally effective, and better than divisions into more or fewer syllables. A syllable length of seven to ten digits seems to be best.

All these points are illustrated in Table II, which compares P_{HW} and P_{SW} for several values of m and $\alpha_{HW} = 1.80$. The table also shows how the syllabified Wagner code finally surpasses the Hamming-Wagner code at about m = 80. As previously mentioned, this is due to the decrease in C_{HW}, the conditional probability that the Hamming-Wagner code corrects double errors, as m increases. This decrease in C_{HW} is also shown in Table II. The formulas used for calculating the P's are the same as those in Eqs. (10), (12), and (13) with the α's related by

$$\alpha_U = \sqrt{\frac{m+k+1}{m}}\,\alpha_{HW} \quad , \quad \alpha_W = \sqrt{\frac{m+k+1}{m+1}}\,\alpha_{HW} \quad , \quad \alpha_{SW} = \sqrt{\frac{m+k+1}{m+j}}\,\alpha_{HW} \tag{18}$$

where m is the number of message digits, k the number of check digits, and j the number of

TABLE II

COMPARISON OF THE HAMMING-WAGNER AND SYLLABIFIED WAGNER CODES				
$a_{HW} = 1.80$				
m	P_{HW}	C_{HW}	j	P_{SW}
12	0.00107	0.77	1	0.0081
			2	0.0087
14	0.00143	0.75	1	0.00148
			2	0.00144
16	0.00186	0.73	2	0.00228
18	0.00236	0.72	2	0.00322
			3	0.00342
20	0.00292	0.70	2	0.00438
			3	0.00449
22	0.00356	0.68	2	0.00577
			3	0.00570
24	0.00426	0.67	3	0.00700
			4	0.00735
30	0.00730	0.62	3	0.00981
			4	0.00975
			5	0.01006
42	0.0146	0.55	5	0.0200
			6	0.0201
54	0.0244	0.50	6	0.0318
			7	0.0317
72	0.0448	0.43	8	0.0468
90	0.0688	0.38	10	0.0659
j = number of syllables				

syllables. The quantity $C_{HW}(a)$ is given by

$$C_{HW}(a) = \frac{2^2 \Pi_n(a)}{n(n-1)\, q^{n-2}(a)\, p^2(a)} \quad , \quad n = m + k + 1 \quad . \tag{19}$$

D. THE REED CODE

We now examine the performance in a constant-data-rate system of the Reed code,[3] the only known example of a systematic multiple-error-correcting code. First we describe the code briefly.

The Reed code is applicable only when the total number of digits in a word is a power of 2. Corresponding to each possible word length, there are only certain possible values of the order to which errors may be corrected. For each of these possible values, the number of message digits is determined. This feature limits the application of the code in communication systems, for the number of message digits in a word (fixed by other considerations) may not

correspond to a possible choice in a Reed code. Table III shows the relations between the number of message digits, the distance between possible transmitted messages (see Appendix A), and the order of errors corrected and detected for Reed-coded words of 2^ν digits.

As indicated in Table III, the Reed code not only corrects all errors up to a given order, but also detects errors of that order plus one. This feature is not an advantage in our case, since there is no indication of the correct replacement for the detected mistaken word. In some cases the Reed code corrects errors of order higher than indicated in Table III,[3] but no

TABLE III

CHARACTERISTICS OF THE REED CODE			
Words of 2^ν digits			
Number of message digits	Distance between possible messages transmitted	Order to which errors are corrected	Order of detected errors
$1 + \nu$	$2^{\nu-1}$	$2^{\nu-2} - 1$	$2^{\nu-2}$
$1 + \nu + \binom{\nu}{2}$	$2^{\nu-2}$	$2^{\nu-3} - 1$	$2^{\nu-3}$
...	...	...	...
$1 + \nu + \binom{\nu}{2} + \ldots + \binom{\nu}{j}$	$2^{\nu-j}$	$2^{\nu-j-1} - 1$	$2^{\nu-j-1}$
...	...	...	...
$2^\nu - 1$	2	0	1
2^ν	1	0	0

analysis of this phenomenon has been made. It follows that the probabilities of error for Reed-coded words calculated below are only upper bounds, albeit probably good ones because of the high order of the extra corrected errors.

Table IV gives a few of the numbers corresponding to the formulas of Table III. The lack of flexibility in the simultaneous choices of number of message digits and order of errors corrected can be seen at once.

The encoded message is obtained by multiplying the message digits by certain standard sequences of $n = 2^\nu$ digits, and then adding the products modulo 2. Decoding is accomplished by choosing that digit given by the majority of a set of sums (again modulo 2), a given standard set corresponding to each message digit. Complete details are to be found in Reed's paper.[3]

Tables III and IV give enough information on the number of check digits, the order of errors corrected, etc. to study the performance of the Reed code in a constant-data-rate system. Table V shows corresponding probabilities of error per word for a Reed three-error-correcting code, the Hamming single-error-correcting code, and no code at all. The formulas for P_U and P_H are the same as those given in Eqs. (10) and (11); the probability P_R is given by

$$P_R = 1 - \sum_{i=0}^{3} \binom{m+k_R}{i} q^{m+k_R-i}(a_R)\, p^i(a_R) \quad , \tag{20}$$

TABLE IV

NUMERICAL EXAMPLES OF THE REED AND HAMMING CODES				
n	m	k_R	k_H	Order to which errors are corrected
8	4	4	3	1
16	5	11	4	3
16	11	5	4	1
32	6	26	4	7
32	16	16	5	3
32	26	6	5	1
64	7	57	4	15
64	22	42	5	7
64	42	22	6	3
64	57	7	6	1
128	99	27	7	3
256	219	37	8	3

TABLE V

PROBABILITIES OF ERROR FOR UNCODED, HAMMING-CODED, AND REED-CODED WORDS				
m	a_H	P_U	P_H	P_R
16	1.5	0.114	0.049	0.047
16	2.0	0.00953	0.00112	0.00042
42	1.5	0.389	0.195	0.163
42	2.0	0.0512	0.0057	0.0011
99	1.5	0.754	0.538	0.449
99	2.0	0.1558	0.0259	0.0041
219	1.5	0.967	0.899	0.839
219	2.0	0.354	0.100	0.018

where k_R is the number of Reed check digits; k_H is the number of Hamming check digits required for m message digits (see Table IV). The relations required to find the α's used in Table V are

$$\alpha_U = \sqrt{\frac{m + k_H}{m}}\,\alpha_H \quad ,$$

$$\alpha_R = \sqrt{\frac{m + k_H}{m + k_R}}\,\alpha_H \quad . \tag{21}$$

We see from Table V that the Reed code outperforms the Hamming code, even for m = 16. Thus the decrease in α produced by the extra check digits of the Reed code is more than compensated by the ability to correct all double and triple errors. The advantage is more marked for larger α, since in high noise many more errors of order greater than three are introduced by the shortening of the digit length.

In Table VI, the Reed code is compared at three of its allowed values of m with the best of the Wagner codes. The probability of error for uncoded words is given for reference.

TABLE VI

COMPARISON OF REED CODE WITH HAMMING-WAGNER AND SYLLABIFIED WAGNER CODES

m	α_{HW}		P_U	P_{HW}	P_R
16	1.80		0.0224	0.0019	0.0022
42	1.80		0.1181	0.0146	0.0097
m	**α_{SW}**	**j**	**P_U**	**P_{SW}**	**P_R**
99	1.50	10	0.726	0.359	0.403
99	2.00	10	0.1379	0.0151	0.0029

We see that the Wagner-type codes can compete with the Reed code in high noise. As the noise decreases or m increases, the Reed code increases its advantage. It is clear that for ordinary communication purposes, the Reed code would be better for long words than any of the previously considered codes if the restriction on the allowed values of m could be removed. Attempts are being made to modify the code to give a greater number of possible message lengths, but as yet no systematic multiple-error-correcting code suitable for an arbitrary number of message digits has been found.

E. SUMMARY AND CONCLUSIONS

We have considered the use of several types of binary codes in communication systems, making the following assumptions:–

(1) The system transmits sequences of binary digits known as words. If any digit is altered, the information carried by a word is lost. Thus, by definition, sequences obtained by combining words are not themselves words.

(2) The transmitted digits are one of two electrical signals of bandwidth W and duration T. They have equal energies and equal a priori probabilities.

(3) The entire coded word must be transmitted in a given time, regardless of the number of code digits required to check the message digits. (Assumption of constant data-rate.)

(4) The transmitted digits are corrupted by the addition of white Gaussian noise. They are determined by choosing the larger of two independent and normally distributed correlator outputs. The time-bandwidth product, TW, of the transmitted signals is $>> 1$, so that when the signal length is changed to accommodate different numbers of check digits, the signal-to-noise ratio of the correlator difference voltage is proportional to the square root of the signal length.[4] (Actually, the signal-to-noise ratio is proportional to $\sqrt{TW}$, but we assume that W is not changed, an assumption that requires $TW >> 1$.)

By the best code (of those we consider) for a given word length and channel noise, we mean that for which the probability of error per word is smallest (under the assumption of constant data-rate). We have considered the following systematic codes: – (1) the Hamming single-error-correcting code, (2) the Wagner code, (3) the Hamming-Wagner code, (4) the syllabified Wagner code, and (5) the Reed multiple-error-correcting codes. (The Wagner, Hamming-Wagner, and syllabified Wagner codes are introduced in this paper.) For short words ($m <$ about 15) we find that the Wagner code is best in the range of interest (neither too little nor too much noise). As m increases, the Wagner code is surpassed by both the Hamming-Wagner code and a syllabified Wagner code of two syllables.* For values of $m <$ about 80, all syllabified Wagner codes are inferior to the Hamming-Wagner code. For larger m, the conditional probability that double errors are corrected by the Hamming-Wagner code has fallen sufficiently so that a syllabified Wagner code is better. Thus, were it not for the Reed code (which is only applicable for a few word lengths), we could say that the Wagner code is best for short words, the Hamming-Wagner code for medium length and long words, and the syllabified Wagner code for very long words. However, the Reed code outperforms the Hamming-Wagner code at $m = 42$ and the syllabified Wagner code at $m = 99$ (except in excessively high noise). Thus for large m there is no substitute for multiple-error-correcting codes that do not use the Wagner principle. We can safely say that if the Reed code can be generalized to apply to any number of message digits, it will be the best code except for short words. This assumes that the proportion of check to message digits turns out to be comparable to that of the present Reed code.

The numerical work reported here was done by Mrs. Elizabeth Munro.

---

*The Hamming code surpasses the Wagner code for m about 20, but is always inferior to the Hamming-Wagner code.

APPENDIX A

The salient features of a geometrical model that is often useful as a visual aid in coding problems are given in the following discussion. This model has already been used to advantage by Hamming.[2]

The set of possible sequences of n binary digits can be represented by the unit cube in a space of n dimensions. The sequences are the vertices of the cube, and the distance between two vertices is defined as the number of binary digits in which the corresponding sequences differ. The set of all points at a distance $\leq k$ from a given point is called the sphere of radius k about that point. The volume of a sphere of radius k, defined as the number of points in the sphere, is

$$\sum_{i=0}^{k} \binom{n}{i} .$$

It can be seen that if we choose any set of mutually exclusive volumes in message space, designate one point in each volume as a possible transmitted message, and identify all other points in the volume with this point, we have constructed a model of an error-correcting code. In particular, if the volumes are spheres of radius k, we have constructed a code that will correct any number of errors $\leq k$.

If the message space is divided into spheres of radius one, we have the geometrical model of the Hamming single-error-correcting code. Unless n is of the form $2^i - 1$, there are points that do not belong to any sphere. (If $n = 2^i - 1$, the space can be fully packed by 2^{n-i} spheres of radius one and volume 2^i.) Points not in any sphere are said to be in limbo. They represent messages that cannot become one of the set of possible transmitted messages by correction of only one digit. For such a point, the Hamming parity checks call for a change in a digit whose order number is greater than the length of the sequence.

Figure 2 illustrates a Hamming single-error-correcting code for a space of $7 = 2^3 - 1$ dimensions. The cube has 128 vertices, and can be fully packed by 16 spheres of radius 1 and volume 8, the centers of which correspond to the 16 possible sequences of 4 binary digits. The centers of the 16 spheres are circled. Note that there are no points not in some sphere.

Figure 3 illustrates the Hamming code for 6 (not of the form $2^i - 1$) dimensions. The centers of the 8 spheres are circled. Note that there are 8 points not in any sphere. These points (enclosed in squares) are the limbo. The dotted outlines in Figs. 2 and 3 indicate typical spheres in each space.

Double-error-correcting codes of dimension less than 90 must have a limbo. For if such a code has no limbo, the volume V of the message sphere (of radius 2) must divide the volume 2^n of the whole space. Thus the equation

$$V = 1 + n + \binom{n}{2} = \frac{1}{2}(n^2 + n + 2) = 2^k \qquad \text{(A-1)}$$

must be satisfied for integral n and k. By inspection, we find that the only solutions of Eq. (A-1) for $n < 90$ are n = 1, 2, and 5. The first two values are meaningless, and the third is the trivial

two-error-correcting code consisting of the two points 00000 and 11111 (i.e., one message digit and four check digits).

The geometrical model appropriate to the Hamming single-error-correcting, double-error-detecting code and the Hamming-Wagner code is obtained by adding an extra dimension to configurations such as those of Figs. 2 and 3. This extra dimension corresponds to the extra check digit required to detect double errors. Some errors involving three or more digits violate this extra parity check and some do not. It is possible to calculate how many errors of each order violate the parity check and how many do not, but this is hardly worth while, since those that violate the check are indistinguishable from double errors and those that do not are indistinguishable from single errors or no errors at all.

Mr. Oliver Selfridge has helped greatly in preparing Appendix A.

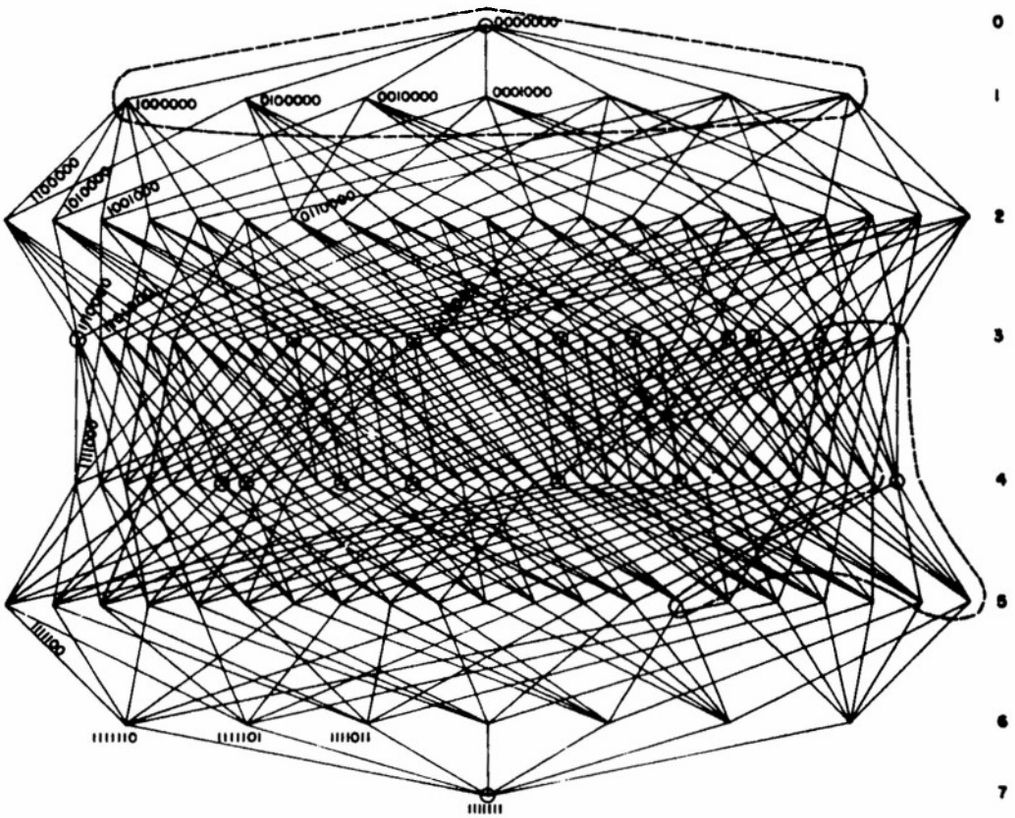


Fig. 2. Space of messages with 7 binary digits.

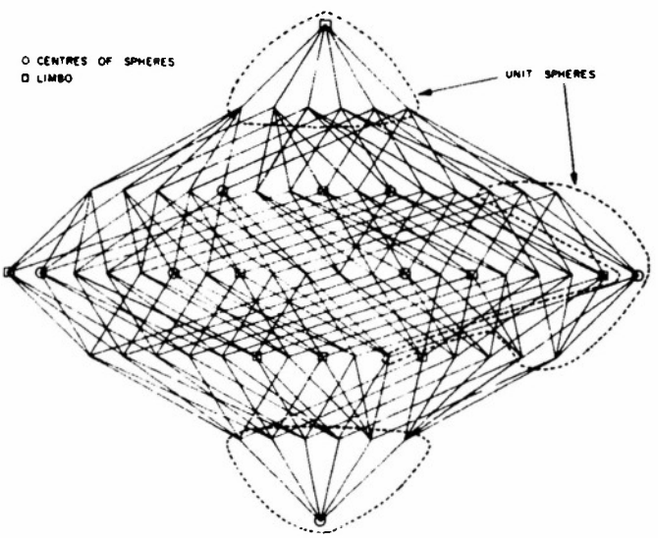


Fig. 3. Space of messages with 6 binary digits.

APPENDIX B

The transition from Eq. (I-22) to Eq. (4) is derived in detail as follows. Equation (I-22) can be written

$$\Pi_n(\alpha) = \sum_{j=1}^{n-1} \frac{(-1)^{n-j+1}}{2^{n-j}} \binom{n}{j} \Pi_j(\alpha) + \frac{(-1)^{n+1}}{2^n} n I_n(\alpha) \quad , \tag{B-1}$$

where we define

$$\Pi_1(\alpha) \equiv \frac{1}{2} I_1(\alpha) \equiv p(\alpha) \quad . \tag{B-2}$$

For n = 2 and n = 3, Eq. (B-1) reduces to

$$\Pi_2(\alpha) = \frac{1}{2} [I_1(\alpha) - I_2(\alpha)]$$

and

$$\Pi_3(\alpha) = \frac{3}{8} [I_1(\alpha) - 2 I_2(\alpha) + I_3(\alpha)] \quad . \tag{B-3}$$

We now prove by induction that

$$\Pi_n(\alpha) = \frac{n}{2^n} \sum_{i=1}^{n} (-1)^{i+1} \binom{n-1}{i-1} I_i(\alpha) \tag{4}$$

is valid for all n. Note first that by Eqs. (B-2) and (B-3), Eq. (4) obtains for n = 1, 2, and 3. Assume that Eq. (4) obtains for all $j \leqslant n-1$. Then

$$\Pi_n(\alpha) = \frac{1}{2^n} \sum_{j=1}^{n-1} \sum_{i=1}^{j} (-1)^{n+i-j} j \binom{n}{j} \binom{j-1}{i-1} I_i(\alpha) + (-1)^{n+1} \frac{n}{2^n} I_n(\alpha)$$

$$= \frac{n}{2^n} \sum_{i=1}^{n-1} \left[\sum_{j=i}^{n-1} (-1)^{n+i-j} \binom{n-1}{j-1} \binom{j-1}{i-1} \right] I_i(\alpha) + (-1)^{n+1} \frac{n}{2^n} I_n(\alpha) \quad . \tag{B-4}$$

Then, it follows from the relation

$$\sum_{j=i}^{n-1} (-1)^{n+i-j} \binom{n-1}{j-1} \binom{j-1}{i-1} = \binom{n-1}{i-i} \sum_{j=i}^{n-1} (-1)^{n+i-j} \binom{n-i}{j-i}$$

$$= \binom{n-1}{i-1} \sum_{r=0}^{n-i-1} (-1)^{n-r} \binom{n-i}{r} = (-1)^{i+1} \binom{n-1}{i-1}$$

that

$$\Pi_n(\alpha) = \frac{n}{2^n} \sum_{i=1}^{n} (-1)^{i+1} \binom{n-1}{i-1} I_i(\alpha) \quad , \tag{4}$$

which completes the induction.

It can be shown in just the same way that Eq. (7) implies Eq. (9).

REFERENCES

1. R.A. Silverman and M. Balser, Technical Report No. 39, Lincoln Laboratory, M.I.T. (23 October 1953).
2. R.W. Hamming, Bell System Tech. Jour. 29, 147 (April 1950).
3. I.S. Reed, Technical Report No. 44, Lincoln Laboratory, M.I.T. (9 October 1953). See also D.E. Muller, Ref. 2 in this report.
4. W.B. Davenport, Jr., R.A. Johnson and D. Middleton, Jour. Appl. Phys. 23, 377 (1952).